

# Implementação de arquiteturas no FPGA DE0 nano Cyclone IV

Prof. Dr. Oscar Eduardo Anacona Mosquera

[oscar.mosquera@ufmt.br](mailto:oscar.mosquera@ufmt.br)

19 de março de 2024

# Conteúdo

## Objetivos

## Introdução

## Arquitetura 01

- Código VHDL
- Família
- Pinagem
- Verilog
- Resultados
- Programação

## Arquitetura 02

## Arquitetura 03

### 1 Objetivos

### 2 Introdução

### 3 Arquitetura 01

- Código VHDL
- Família
- Pinagem
- Verilog
- Resultados
- Programação

### 4 Arquitetura 02

### 5 Arquitetura 03

# Objetivos

## Objetivos

### Introdução

#### Arquitetura 01

Código VHDL

Família

Pinagem

Verilog

Resultados

Programação

#### Arquitetura 02

#### Arquitetura 03

### ● Arquitetura da DE0 Nano:

- ▶ Compreender a arquitetura da placa DE0 Nano, incluindo seus componentes principais e especificações técnicas.

### ● Implementação de Vários Exemplos em VHDL e no FPGA:

- ▶ Desenvolver habilidades práticas na escrita de código VHDL para implementar diferentes exemplos de circuitos digitais.
- ▶ Explorar o processo de implementação desses exemplos no FPGA DE0 Nano e testá-los em hardware.

### ● Configuração da Família de FPGA:

- ▶ Entender os procedimentos de configuração da família de FPGA, especialmente do FPGA presente na DE0 Nano.

### ● Configuração da Pinagem do FPGA:

- ▶ Aprender a configurar a pinagem do FPGA para garantir a conectividade correta entre os dispositivos externos e o FPGA na DE0 Nano.

### ● Máquina de Estados Finitos no Quartus:

- ▶ Adquirir conhecimento sobre a implementação de máquinas de estados finitos no software Quartus Prime.
- ▶ Explorar as ferramentas e recursos disponíveis no Quartus para desenvolver e testar máquinas de estados finitos.

# Conteúdo

Objetivos

Introdução

Arquitetura 01

Código VHDL

Família

Pinagem

Verilog

Resultados

Programação

Arquitetura 02

Arquitetura 03

## 1 Objetivos

## 2 Introdução

## 3 Arquitetura 01

- Código VHDL
- Família
- Pinagem
- Verilog
- Resultados
- Programação

## 4 Arquitetura 02

## 5 Arquitetura 03

# Arquitetura da placa DE0 nano

Objetivos

Introdução

Arquitetura 01

Código VHDL

Família

Pinagem

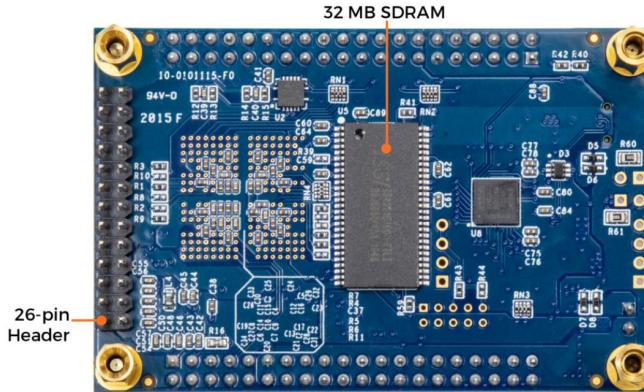
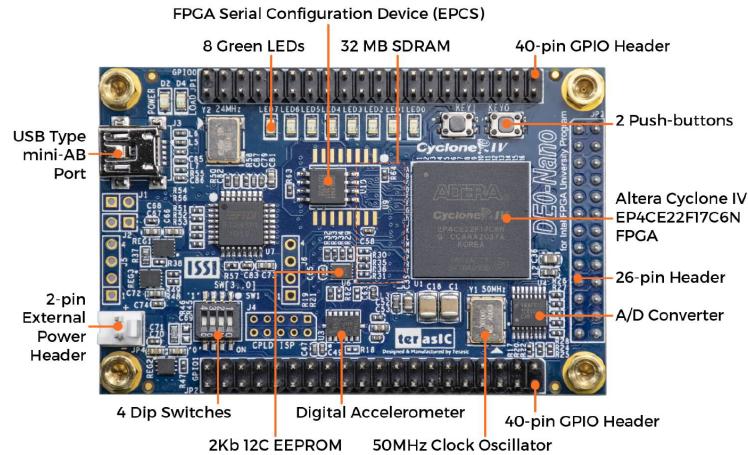
Verilog

Resultados

Programação

Arquitetura 02

Arquitetura 03



# Arquitetura da placa DE0 nano

Objetivos

Introdução

Arquitetura 01

Código VHDL

Família

Pinagem

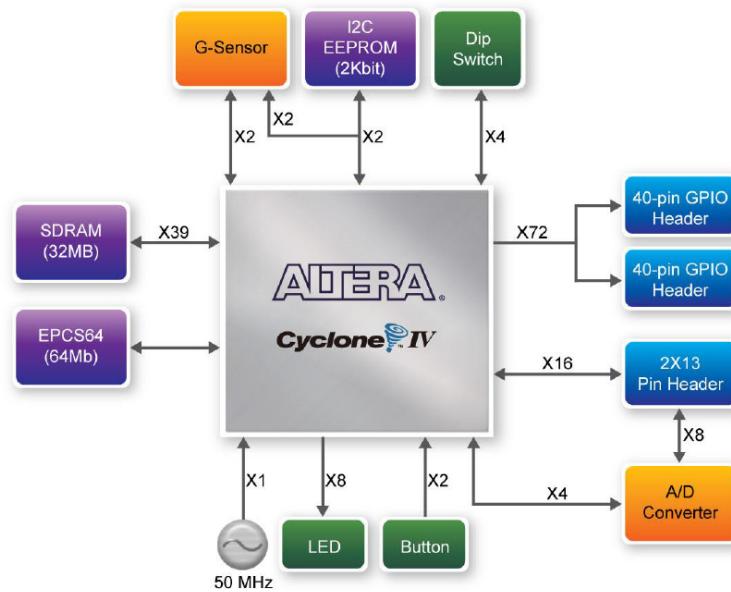
Verilog

Resultados

Programação

Arquitetura 02

Arquitetura 03



# Conteúdo

Objetivos

Introdução

Arquitetura 01

Código VHDL

Família

Pinagem

Verilog

Resultados

Programação

Arquitetura 02

Arquitetura 03

## 1 Objetivos

## 2 Introdução

## 3 Arquitetura 01

- Código VHDL
- Família
- Pinagem
- Verilog
- Resultados
- Programação

## 4 Arquitetura 02

## 5 Arquitetura 03

# Implementação de uma arquitetura no FPGA

Objetivos

Introdução

Arquitetura 01

Código VHDL

Família

Pinagem

Verilog

Resultados

Programação

Arquitetura 02

Arquitetura 03

```

LIBRARY IEEE;
USE IEEE.STD_LOGIC_1164.ALL;

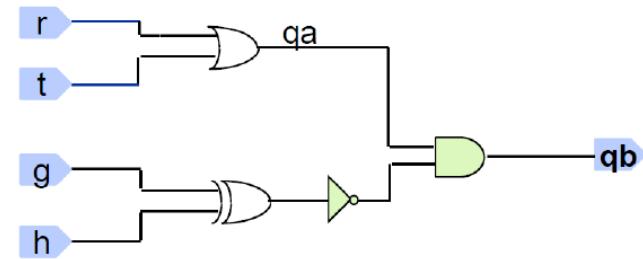
ENTITY simp IS
  PORT (
    r, t, g, h : IN STD_LOGIC;
    qb : OUT STD_LOGIC
  );
END ENTITY simp;

ARCHITECTURE logic OF simp IS
  SIGNAL qa : STD_LOGIC; ←
BEGIN

  qa <= r OR t;
  qb <= (qa AND NOT (g XOR h));

END ARCHITECTURE logic;

```



*Signal declaration  
inside architecture*

- r, t, g, h, and qb are signals (by default)
- qa is a buried signal and needs to be declared

# Implementação de uma arquitetura no FPGA

Objetivos

Introdução

Arquitetura 01

Código VHDL

Família

Pinagem

Verilog

Resultados

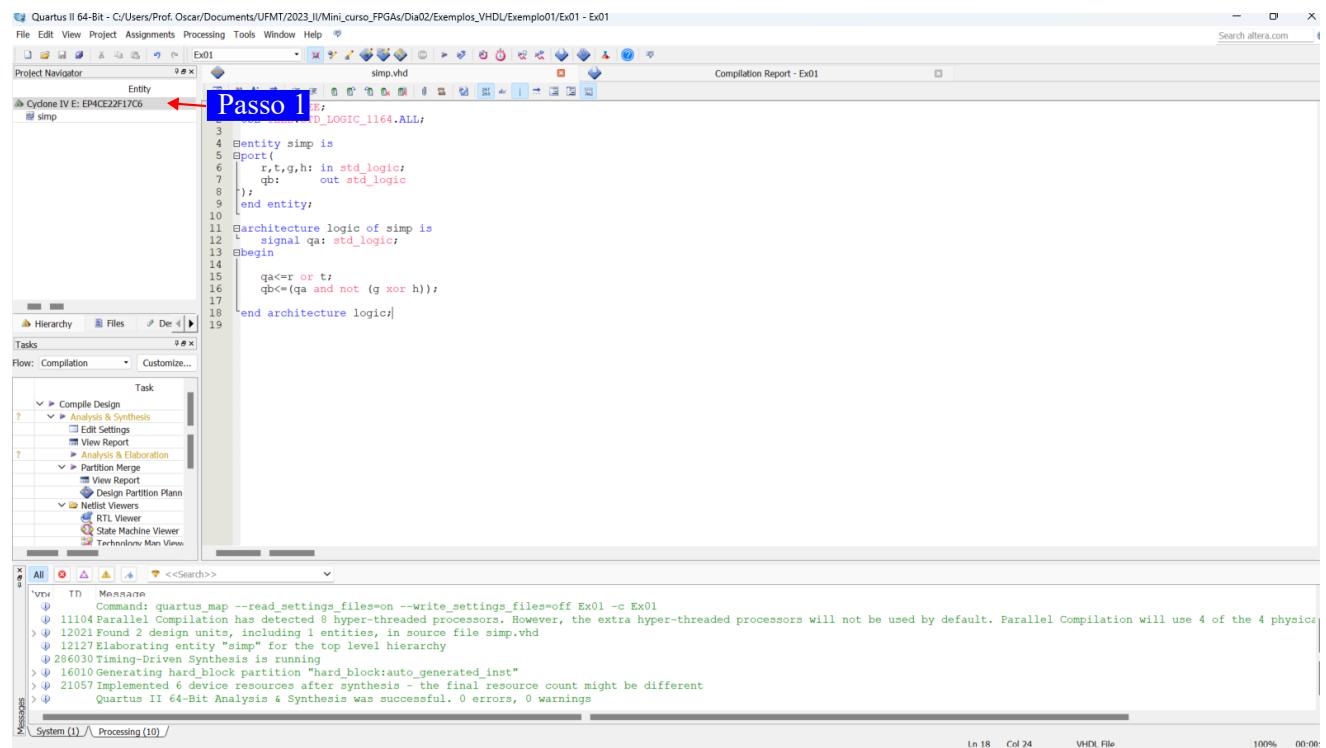
Programação

Arquitetura 02

Arquitetura 03

## Configuração da família da placa a ser usada

- Passo 1: click direito na família



# Implementação de uma arquitetura no FPGA

## Objetivos

## Introdução

## Arquitetura 01

Código VHDL

### Família

Pinagem

Verilog

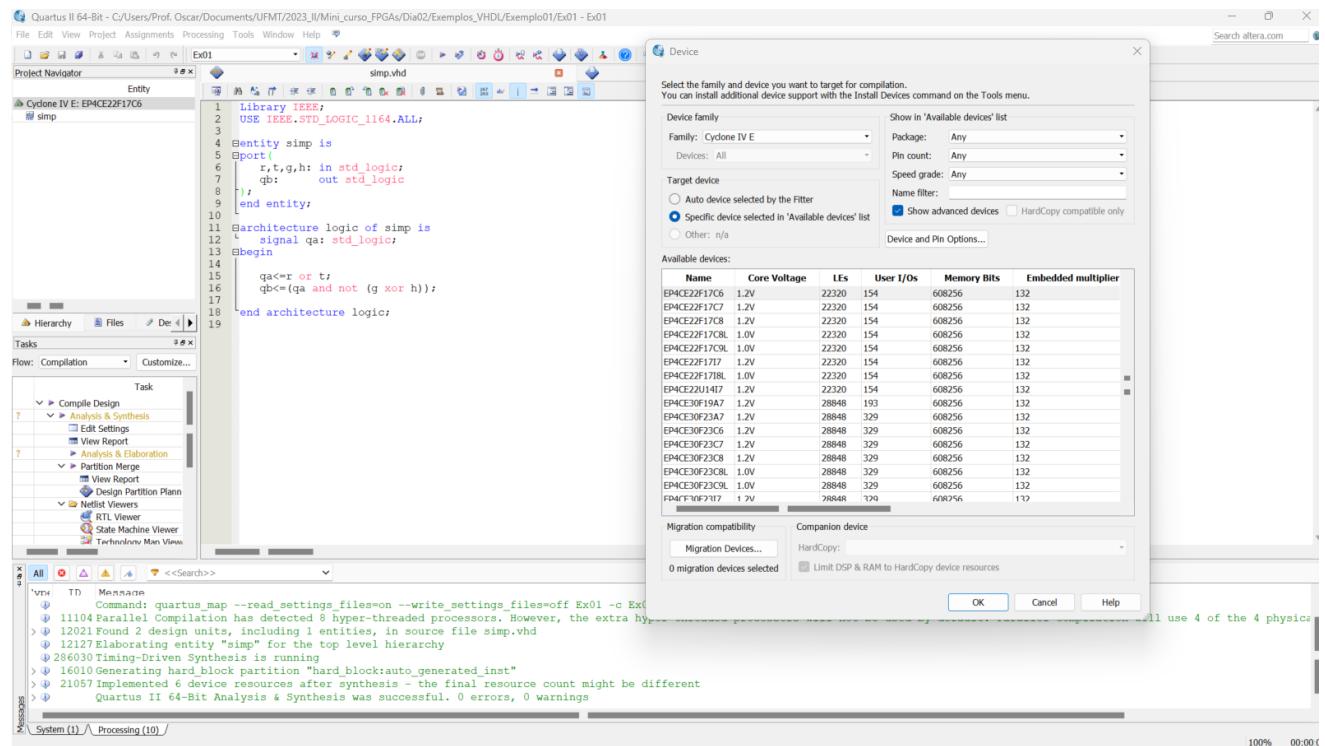
Resultados

Programação

## Arquitetura 02

## Arquitetura 03

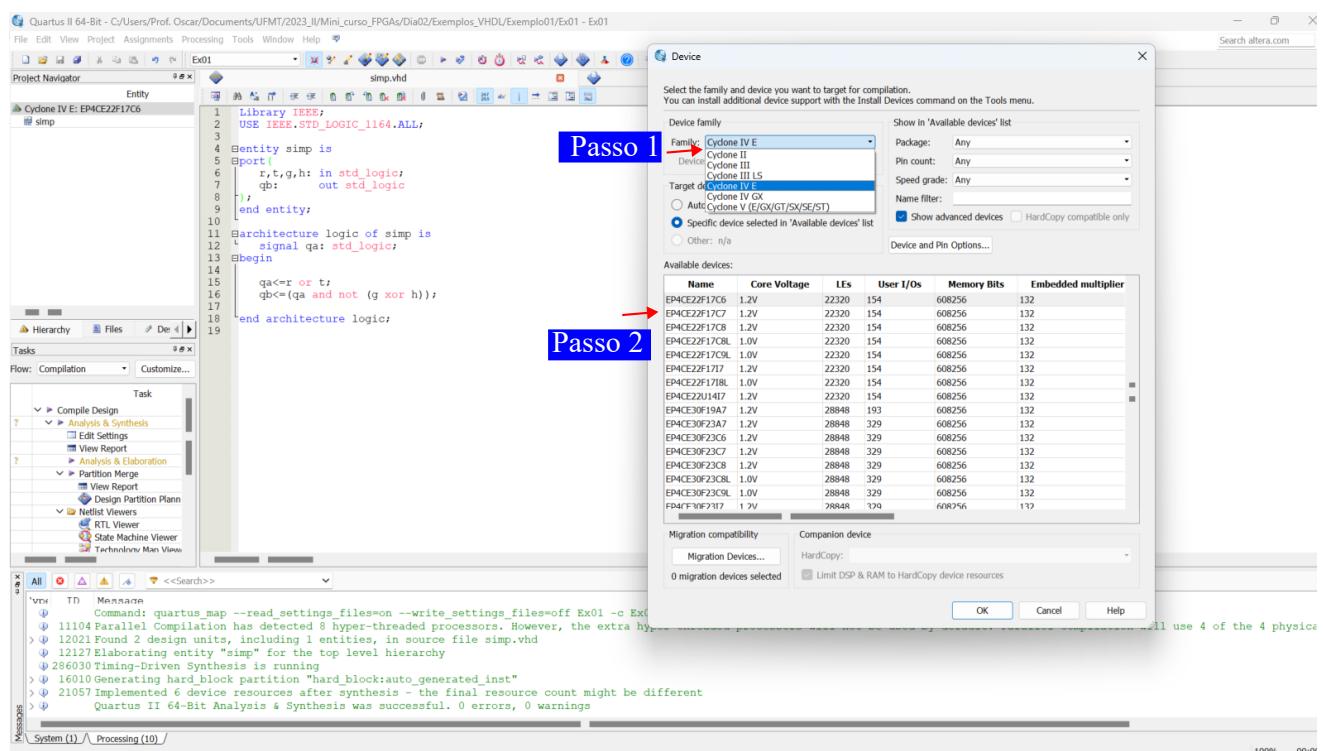
## Configuração da família da placa a ser usada



# Implementação de uma arquitetura no FPGA

## Configuração da família da placa a ser usada

- **Passo 1:** escolher a família
- **Passo 2:** escolher o modelo da placa



# Implementação de uma arquitetura no FPGA

## Objetivos

### Introdução

### Arquitetura 01

Código VHDL

#### Família

Pinagem

Verilog

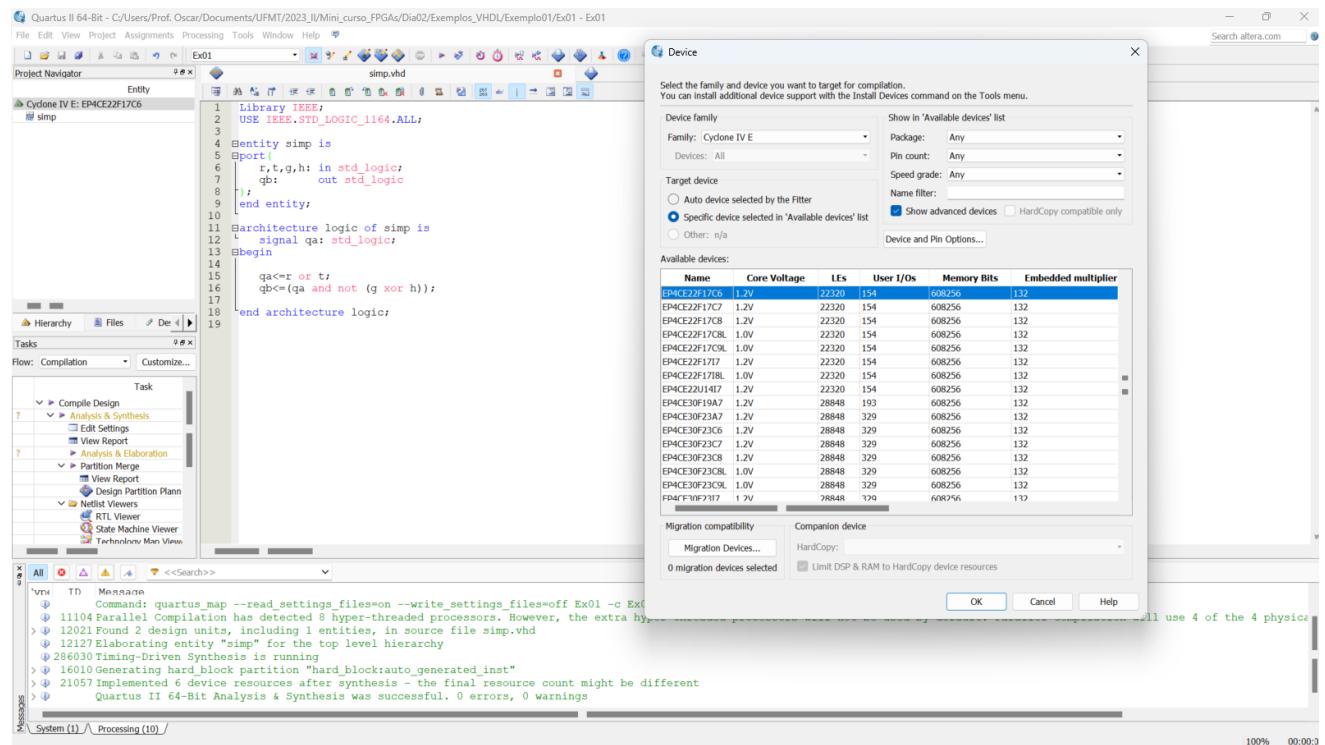
Resultados

Programação

### Arquitetura 02

### Arquitetura 03

## Configuração da família da placa a ser usada



# Implementação de uma arquitetura no FPGA

Objetivos

Introdução

Arquitetura 01

Código VHDL

Família

Pinagem

Verilog

Resultados

Programação

Arquitetura 02

Arquitetura 03

Quartus II 64-Bit - C:\Users\Prof. Oscar\Documents\UFMT/2023\_II\Mini\_curso\_FPGAs\Dia02\Exemplos\_VHDL\Exemplo01\Ex01

File Edit View Project Assignments Processing Tools Window Help

Cyclone IV E: EP4CE22F17C6

Project Navigator Entity simp

```

1 Library IEEE;
2 USE IEEE.STD_LOGIC_1164.ALL;
3
4 Entity simp is
5 Port (
6   r,t,g,h: in std_logic;
7   qb:        out std_logic
8 );
9 end entity;
10
11 Architecture logic of simp is
12   signal qa: std_logic;
13 begin
14
15   qa<=r or t;
16   qb<=(qa and not (g xor h));
17
18 end architecture logic;
19

```

Hierarchy Files Details Tasks Flow: Compilation Customize...

Task

- Compile Design
- Analysis & Synthesis
  - Edit Settings
  - View Report
- Analysis & Elaboration
- Partition Merge
  - View Report
  - Design Partition Planner
- Netlist Viewers
  - RTL Viewer
  - State Machine Viewer
  - Technician View

All TD Message

Command: quartus\_map --read\_settings\_files=on --write\_settings\_files=off Ex01 -c Ex01

11104 Parallel Compilation has detected 8 hyper-threaded processors. However, the extra hyper-threaded processors will not be used by default. Parallel Compilation will use 4 of the 4 physical processors.

112021 Found 2 design units, including 1 entities, in source file simp.vhd

12127 Elaborating entity "simp" for the top level hierarchy

286030 Timing-Driven Synthesis is running

16010 Generating hard\_block partition "hard\_block:auto\_generated\_inst"

21057 Implemented 6 device resources after synthesis - the final resource count might be different

Quartus II 64-Bit Analysis & Synthesis was successful. 0 errors, 0 warnings

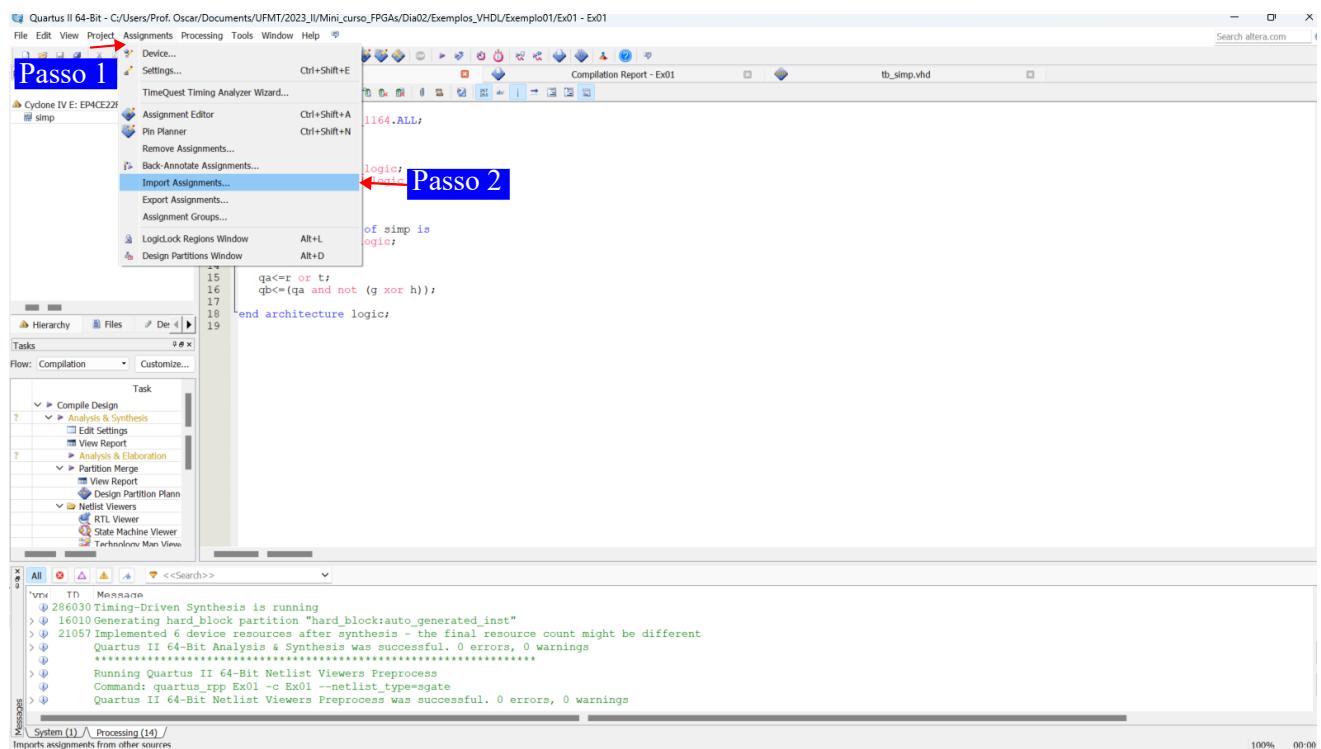
System (1) / Processing (10) /

100% 00:00:0

# Implementação de uma arquitetura no FPGA

## Adição do arquivo com a configuração dos pinos do FPGA

- **Passo 1:** Assignments
- **Passo 2:** Import Assignments



# Implementação de uma arquitetura no FPGA

Objetivos

Introdução

Arquitetura 01

Código VHDL

Família

Pinagem

Verilog

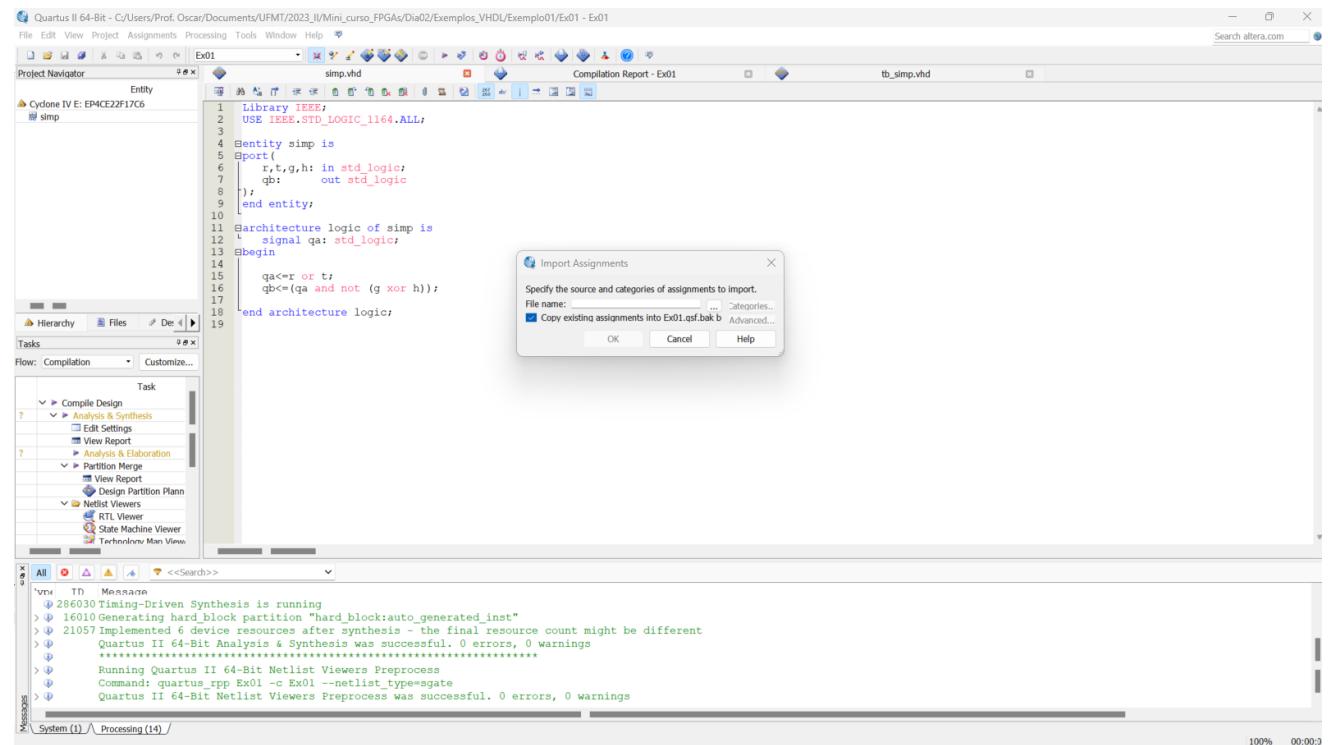
Resultados

Programação

Arquitetura 02

Arquitetura 03

## Adição do arquivo com a configuração dos pinos do FPGA: procurar o arquivo .qsf



# Implementação de uma arquitetura no FPGA

Objetivos

Introdução

Arquitetura 01

Código VHDL

Família

Pinagem

Verilog

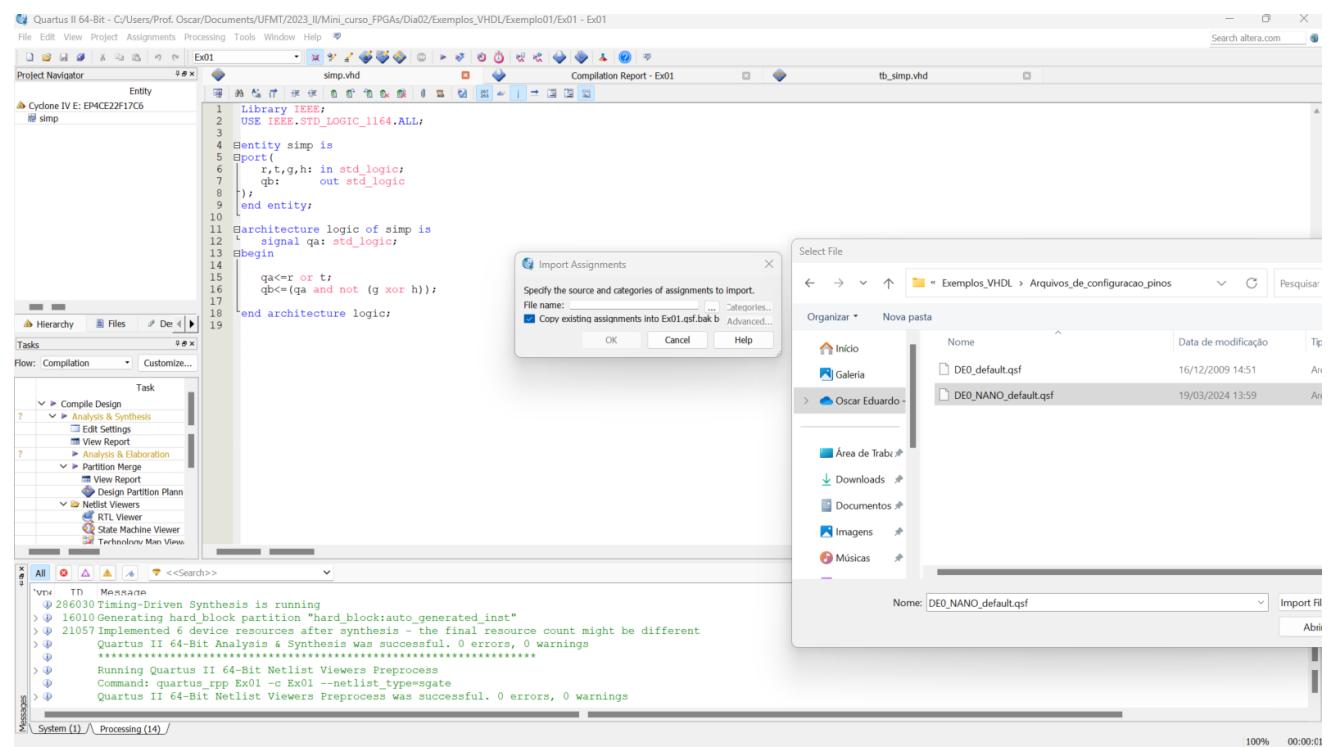
Resultados

Programação

Arquitetura 02

Arquitetura 03

## Adição do arquivo com a configuração dos pinos do FPGA: procurar o arquivo .qsf



# Implementação de uma arquitetura no FPGA

Objetivos

Introdução

Arquitetura 01

Código VHDL

Família

Pinagem

Verilog

Resultados

Programação

Arquitetura 02

Arquitetura 03

## Adição do arquivo com a configuração dos pinos do FPGA: procurar o arquivo .qsf

The screenshot shows the Quartus II 64-Bit software interface. The main window displays a VHDL code for a logic element (simp.vhd) with the following content:

```

1 Library IEEE;
2 USE IEEE.STD_LOGIC_1164.ALL;
3
4 Entity simp is
5 Port(
6   r,t,g,h: in std_logic;
7   qb:        out std_logic
8 );
9 End Entity;
10
11 Architecture logic of simp is
12 Signal qa: std_logic;
13
14 Begin
15
16   qa<=r or t;
17   qb<=(qa and not (g xor h));
18 End Architecture logic;
19

```

The left side of the interface shows a tree view of the project tasks, including options like Compile Design, Analysis & Synthesis, Partition Merge, and Netlist Viewers.

The bottom pane shows a 'Messages' window with the following log output:

```

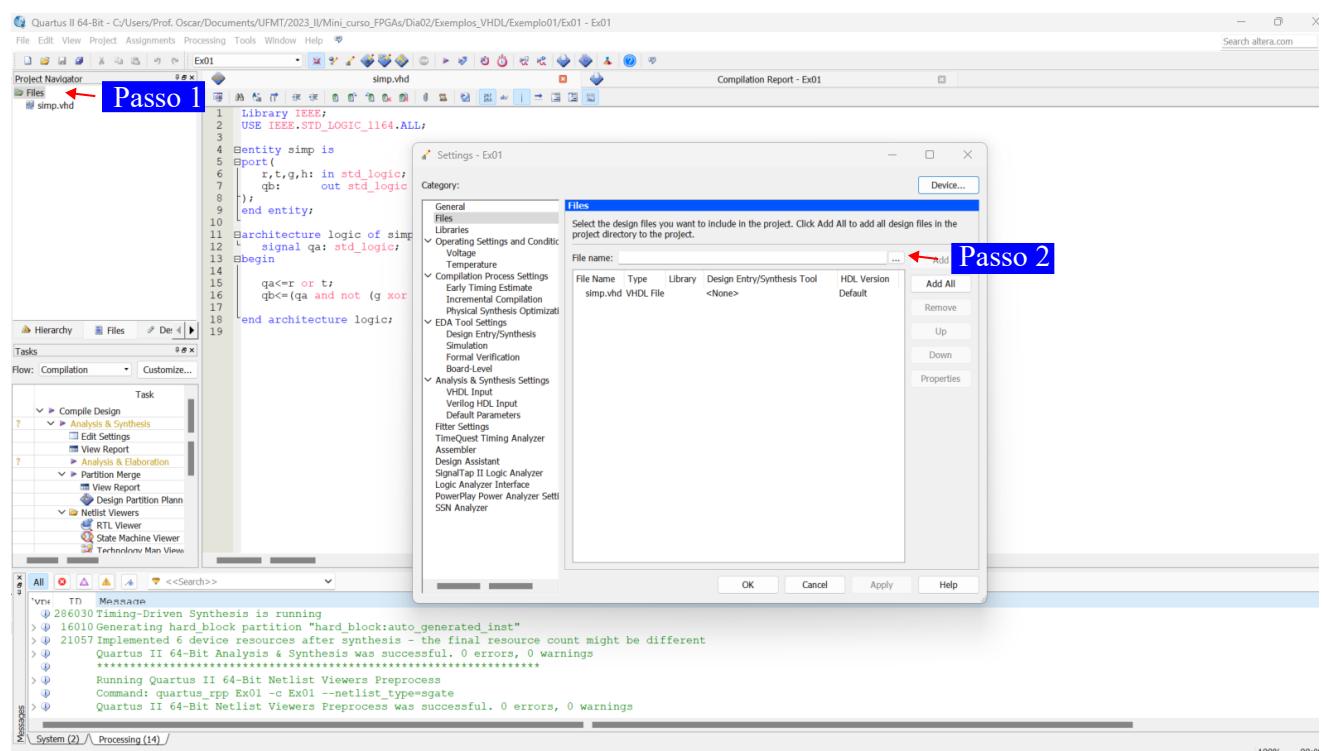
* * * * * Timing-Driven Synthesis is running
> 286030 Generating hard-block partition "hard-blockauto_generated_inst"
> 16010 Generating hard-block partition "hard-blockauto_generated_inst"
> 21057 implemented & 0 unimplemented after synthesis - the final resource count might be different
> Quartus II 64-Bit Analysis & Synthesis was successful. 0 errors, 0 warnings
* * * * *
> Running Quartus II 64-Bit Netlist Viewers Preprocess
> Command: quartus_rpp Ex01 -c Ex01 --netlist_type=gate
> Quartus II 64-Bit Netlist Viewers Preprocess was successful. 0 errors, 0 warnings

```

# Implementação de uma arquitetura no FPGA

## Adição do arquivo em verilog com a lista de pinos do FPGA

- **Passo 1:** Files
- **Passo 2:** ..., depois procurar os arquivos na pasta de trabalho



# Implementação de uma arquitetura no FPGA

Objetivos

Introdução

Arquitetura 01

Código VHDL

Família

Pinagem

Verilog

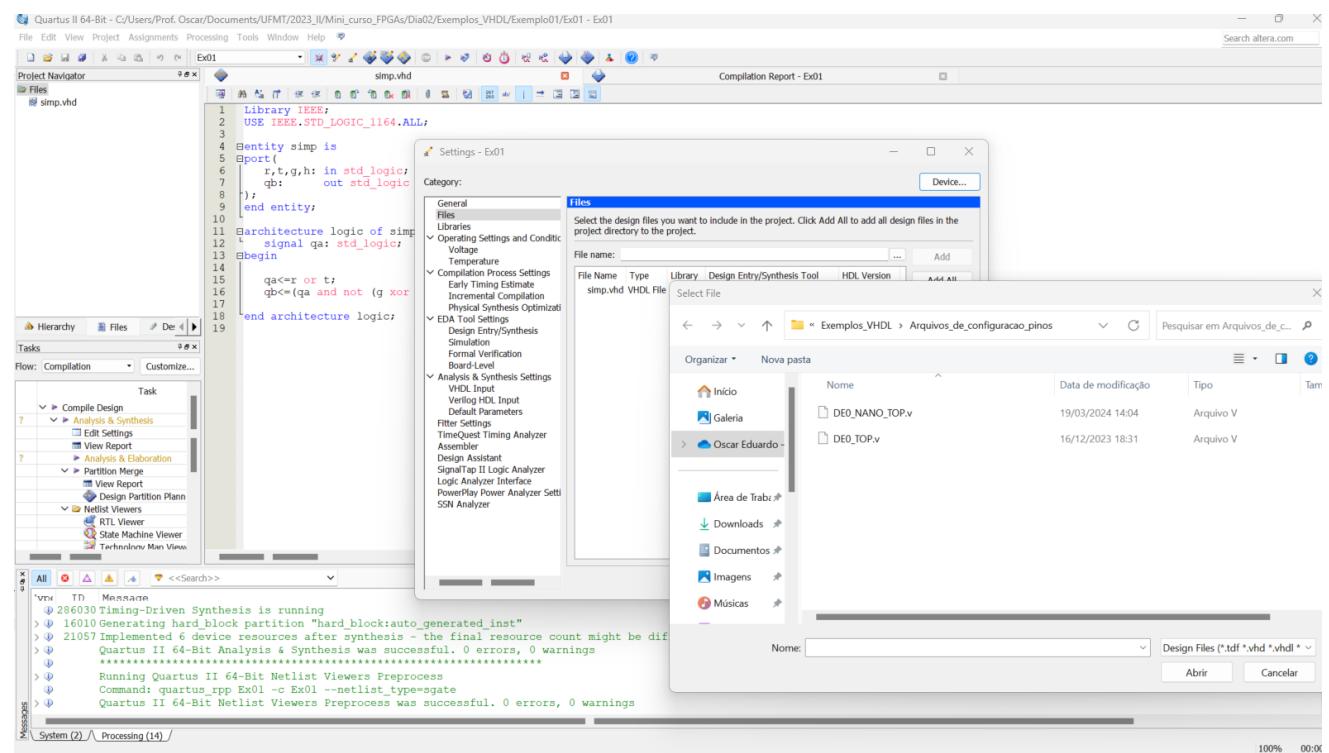
Resultados

Programação

Arquitetura 02

Arquitetura 03

## Adição do arquivo em verilog com a lista de pinos do FPGA



# Implementação de uma arquitetura no FPGA

Objetivos

Introdução

Arquitetura 01

Código VHDL

Família

Pinagem

Verilog

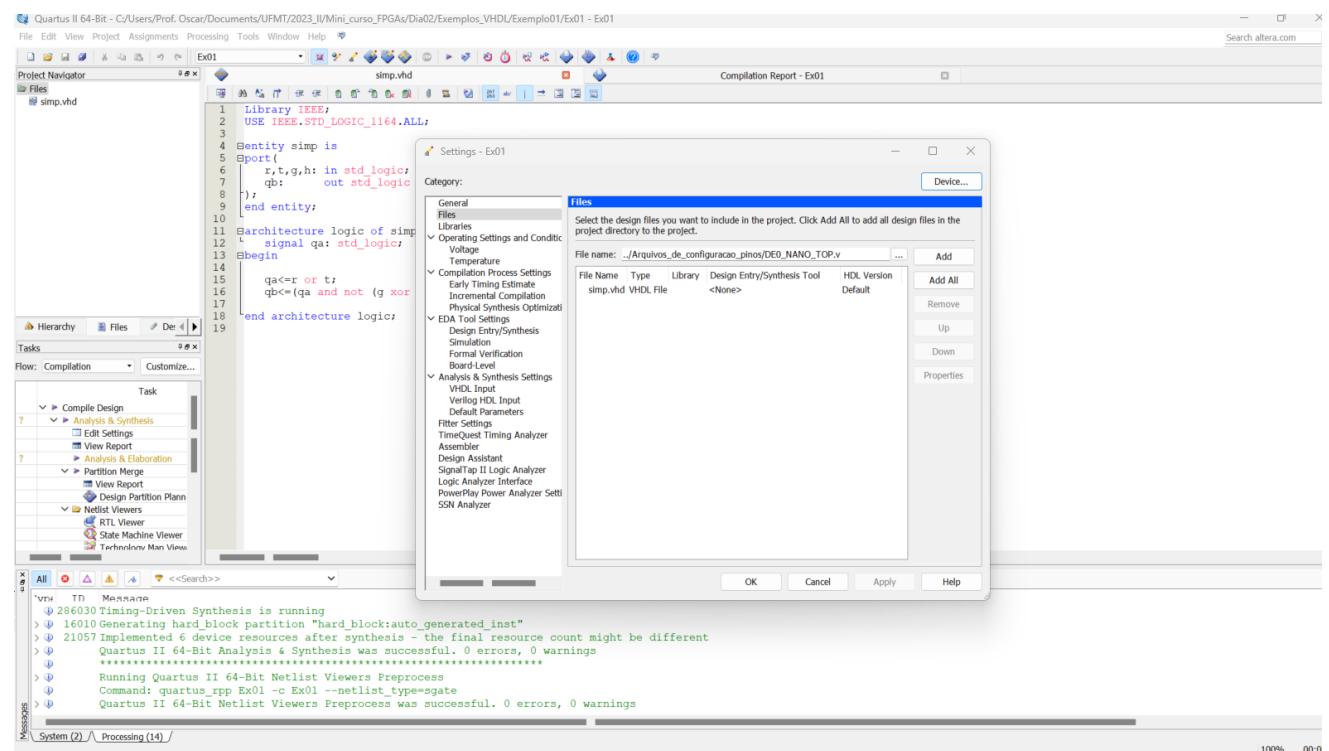
Resultados

Programação

Arquitetura 02

Arquitetura 03

## Adição do arquivo em verilog com a lista de pinos do FPGA



# Implementação de uma arquitetura no FPGA

Objetivos

Introdução

Arquitetura 01

Código VHDL

Família

Pinagem

Verilog

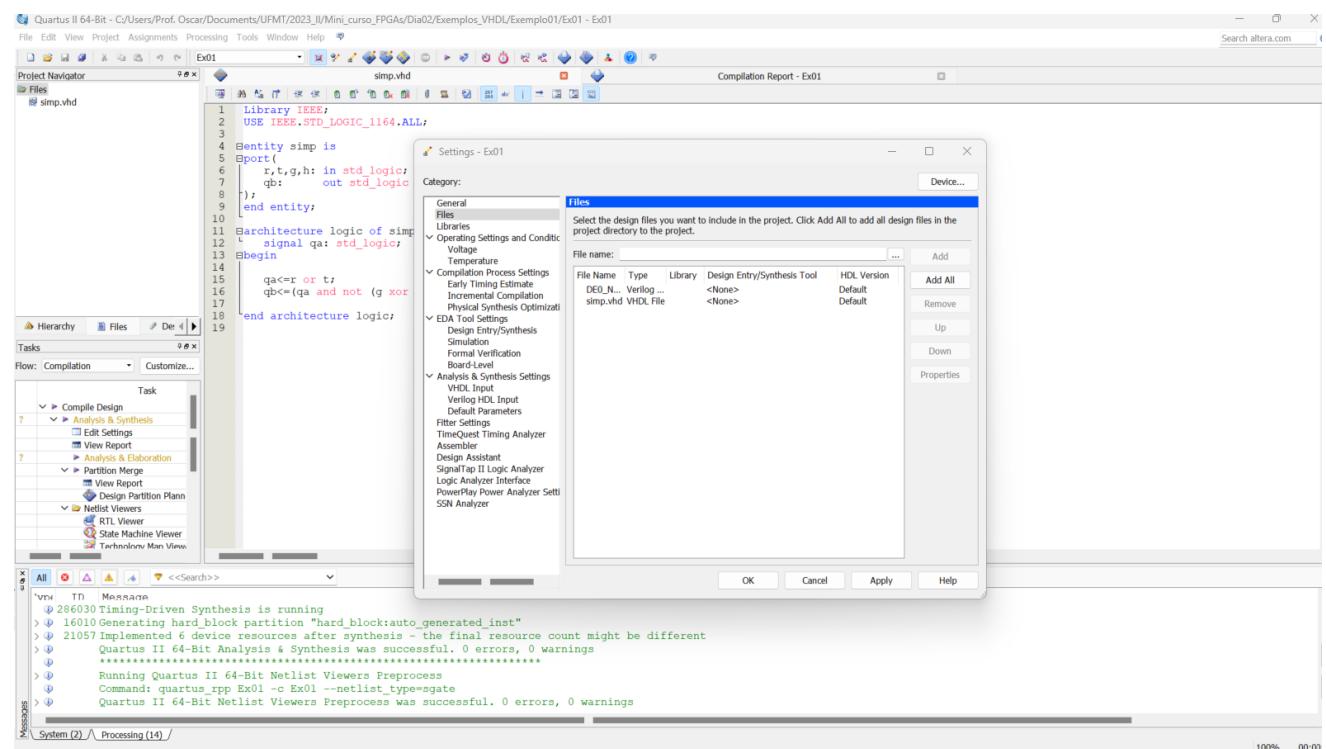
Resultados

Programação

Arquitetura 02

Arquitetura 03

## Adição do arquivo em verilog com a lista de pinos do FPGA



# Implementação de uma arquitetura no FPGA

Objetivos

Introdução

Arquitetura 01

Código VHDL

Família

Pinagem

Verilog

Resultados

Programação

Arquitetura 02

Arquitetura 03

## Arquivo em verilog como invólucro para ligar os pinos do FPGA com a arquitetura

The screenshot shows the Quartus II 64-Bit interface with the following details:

- Title Bar:** Quartus II 64-Bit - C:\Users\Prof. Oscar\Documents\UFMT\2023\_II\Mini\_curso\_FPGAs\Dia02\Exemplos\_VHDL\Exemplo01\Ex01 - Ex01
- Toolbar:** File, Edit, View, Project, Assignments, Processing, Tools, Window, Help.
- Project Navigator:** Shows files: simp.vhd, DE0\_NANO\_TOP.v
- Code Editor:** Displays Verilog code for a DE0\_NANO\_TOP design. The code includes declarations for CLOCK, LED, KEY, and SW inputs, followed by structural coding and an instantiation of a module named 'simp'.

```

// PORT declarations
// -----
// input      CLOCK_50;
// output     [1:0] LED;
// input      [1:0] KEY;
// input      [3:0] SW;

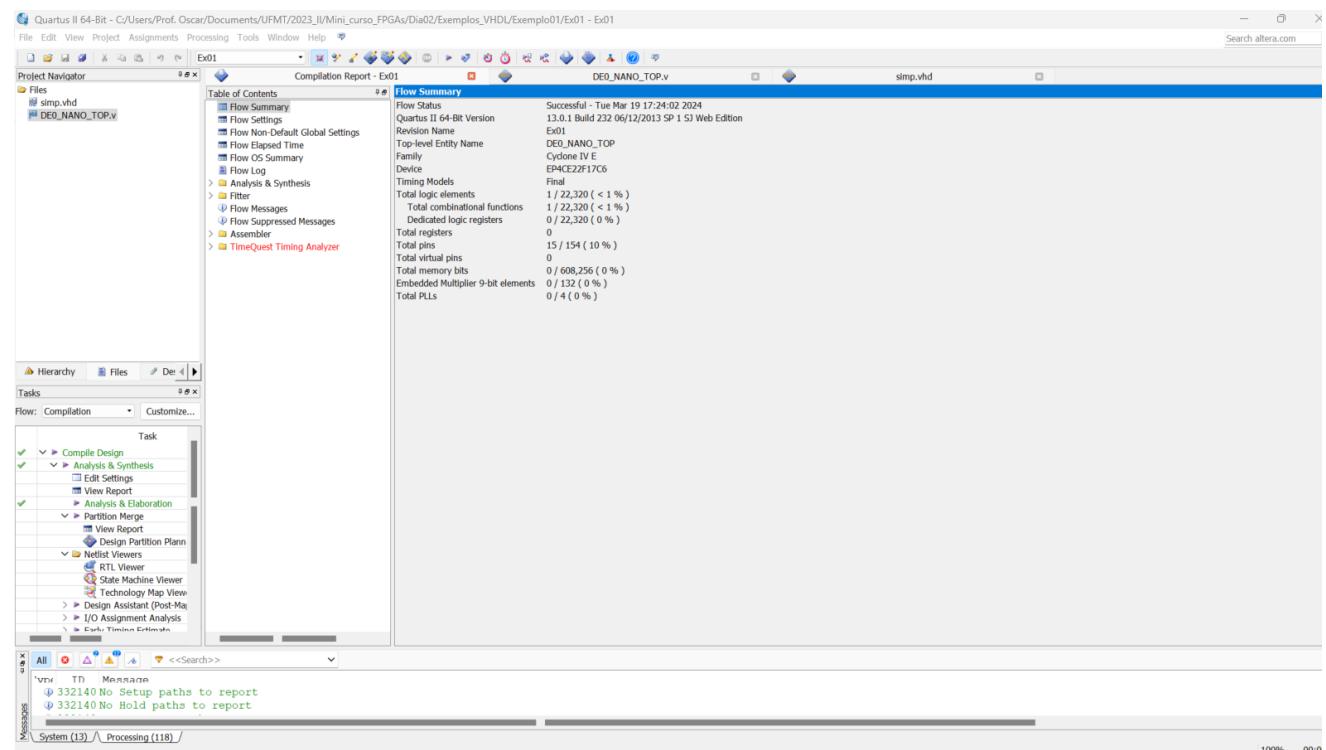
// REG/WIRE declarations
// -----
// ===== Structural coding =====
// -----
// ===== Instantiation =====
simp inst1(
    .r(SW[0]),
    .t(SW[1]),
    .g(SW[2]),
    .h(SW[3]),
    .p(LED[1])
);
endmodule

```

- Task List:** Shows tasks under 'Compilation' flow: Compile Design, Analysis & Synthesis, Analysis & Elaboration, Partition Merge, Netlist Viewers, RTL Viewer, Schematic Viewer, Technology Map View, Design Assistant (Post-Mapping), I/O Assignment Analysis, and Configuration File.
- Messages:** Shows messages related to setup and hold paths.
- Status Bar:** System (13) / Processing (118) / 100% 00:00

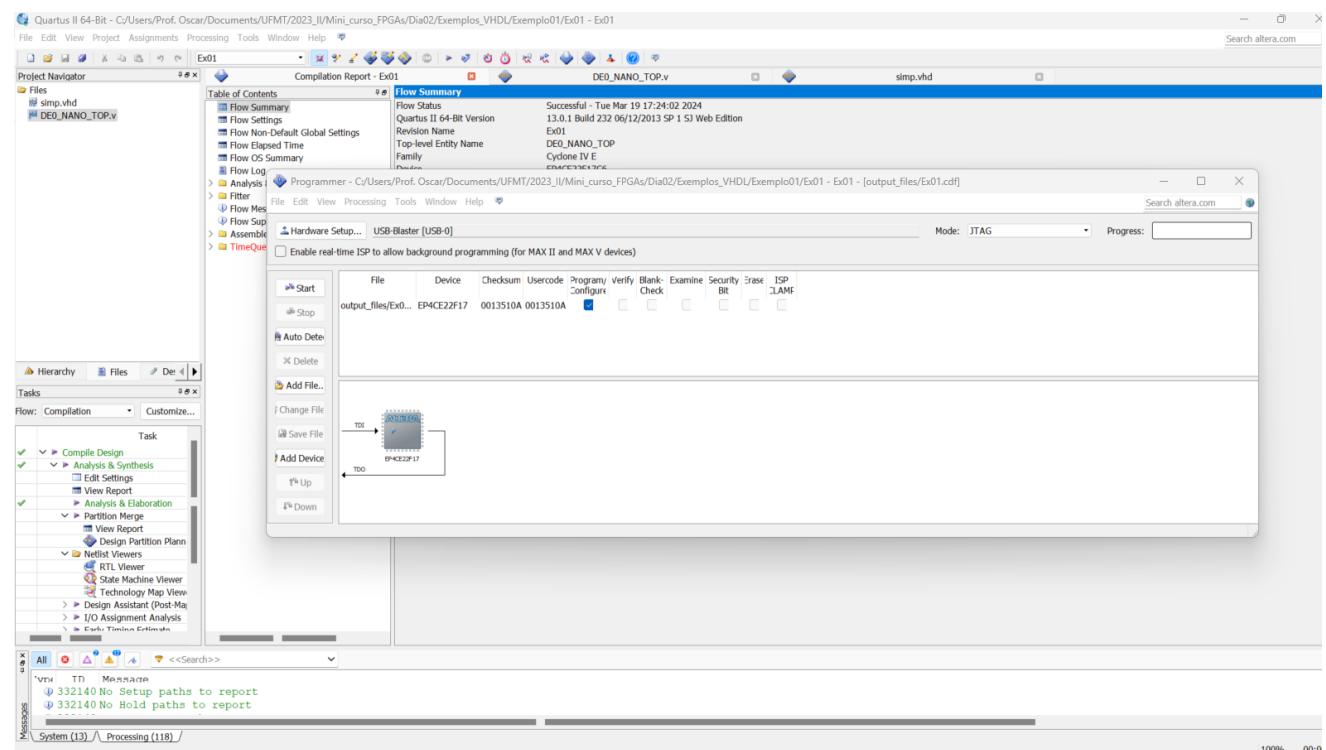
# Implementação de uma arquitetura no FPGA

## Lista de Elementos Lógicos



# Implementação de uma arquitetura no FPGA

## Programação da placa



# Implementação de uma arquitetura no FPGA

Objetivos

Introdução

Arquitetura 01

Código VHDL

Família

Pinagem

Verilog

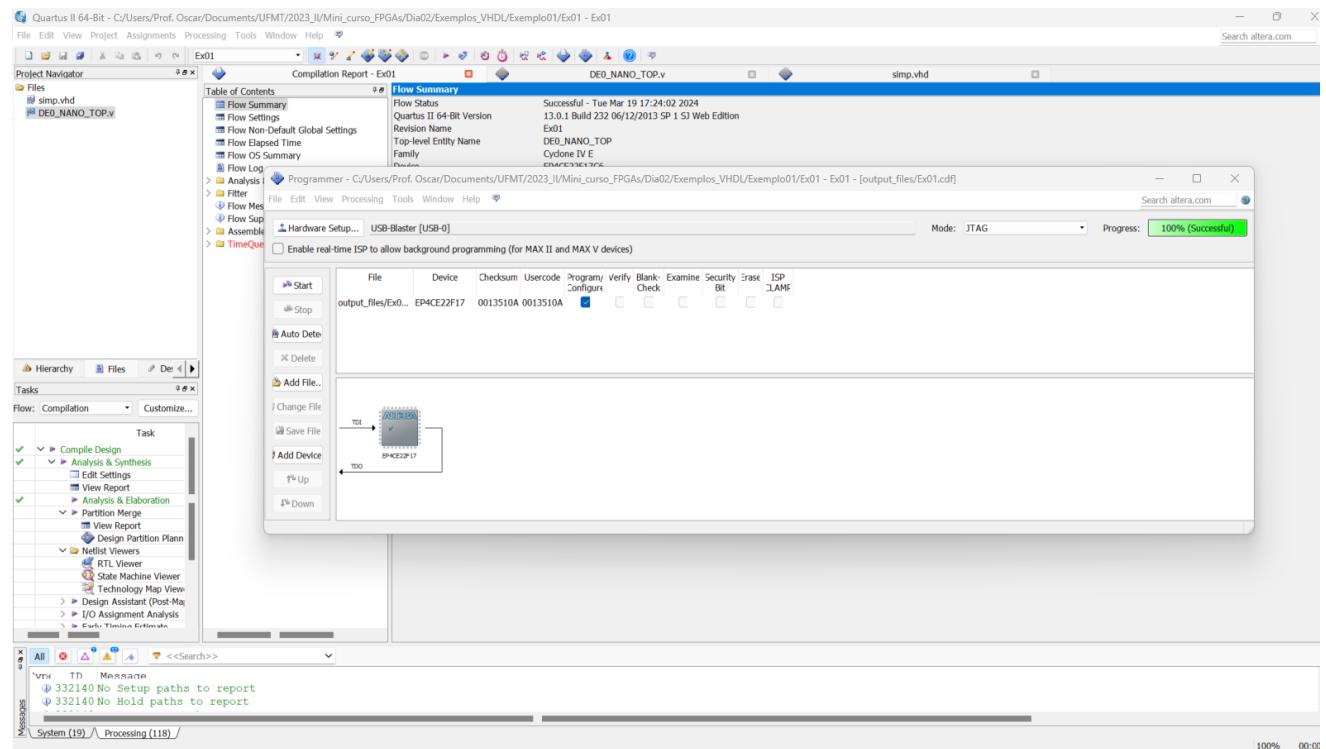
Resultados

Programação

Arquitetura 02

Arquitetura 03

## Programação da placa



# Conteúdo

Objetivos

Introdução

Arquitetura 01

Código VHDL  
Família  
Pinagem  
Verilog  
Resultados  
Programação

Arquitetura 02

Arquitetura 03

## 1 Objetivos

## 2 Introdução

## 3 Arquitetura 01

- Código VHDL
- Família
- Pinagem
- Verilog
- Resultados
- Programação

## 4 Arquitetura 02

## 5 Arquitetura 03

# Implementação de uma arquitetura no FPGA

Objetivos

Introdução

Arquitetura 01

Código VHDL

Família

Pinagem

Verilog

Resultados

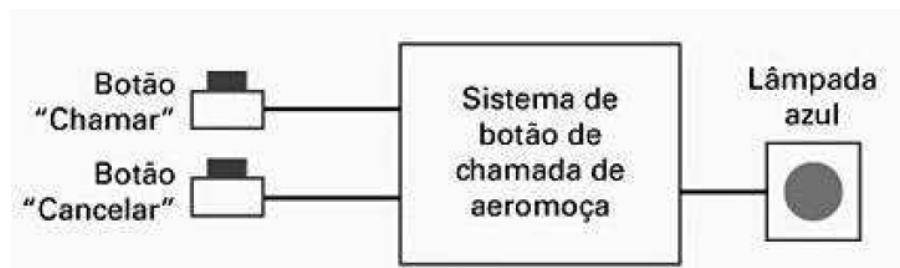
Programação

Arquitetura 02

Arquitetura 03

## Botão de chamada de aeromoça usando um flip-flop D

Implementar um sistema de botão de chamada de aeromoça. Quando o botão **Chamar** é pressionado, a luz acende-se e mantém-se assim depois que **chamar** é solto. Quando o botão **cancelar** é pressionado, a luz apaga-se.



# Conteúdo

Objetivos

Introdução

Arquitetura 01

Código VHDL

Família

Pinagem

Verilog

Resultados

Programação

Arquitetura 02

Arquitetura 03

## 1 Objetivos

## 2 Introdução

## 3 Arquitetura 01

- Código VHDL
- Família
- Pinagem
- Verilog
- Resultados
- Programação

## 4 Arquitetura 02

## 5 Arquitetura 03

# Implementação de uma arquitetura no FPGA

Objetivos

Introdução

Arquitetura 01

Código VHDL

Família

Pinagem

Verilog

Resultados

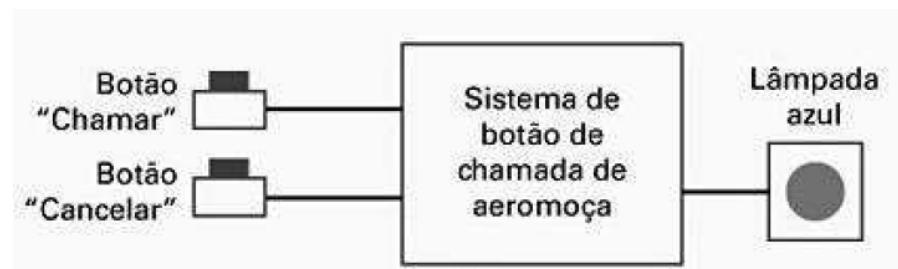
Programação

Arquitetura 02

Arquitetura 03

## Botão de chamada de aeromoça usando um flip-flop D

Implementar um sistema de botão de chamada de aeromoça. Quando o botão **Chamar** é pressionado, a luz acende-se e mantém-se assim depois que **chamar** é solto. Quando o botão **cancelar** é pressionado, a luz apaga-se. Projete o circuito usando uma **Máquina de Estados Finita (FSM)**.



# Implementação de uma arquitetura no FPGA

Objetivos

Introdução

Arquitetura 01

Código VHDL

Família

Pinagem

Verilog

Resultados

Programação

Arquitetura 02

Arquitetura 03

